

文章编号: 1000-8055(2004)06-0776-06

# 一种全程控制的模糊遗传算法在结构优化中的应用

郭海丁, 李湘萍

(南京航空航天大学 能源与动力学院, 江苏 南京 210016)

**摘要:** 将一种全程控制的模糊遗传算法(FGA 算法)引入结构优化设计。区别于一般的标准遗传算法(SGA 算法), 该算法基于模糊推断机理, 可对遗传算法的选择、交叉、变异以及搜索空间的变化进行全程控制。通过 2 个典型数值多峰函数对 FGA 算法与 SGA 算法性能进行了考核和对比, 证明该算法在跳出局部最优和搜索效率等方面均有较大改进。将此改进的模糊遗传算法(FGA 算法)应用于含整型和离散变量的铆钉连接结构连接效率优化。结果表明: 连接效率和优化效率均得到改善。

**关键词:** 航空、航天推进系统; 结构优化; 模糊遗传算法; 全程控制

**中图分类号:** TH 122

**文献标识码:** A

## Structure Optimization Based on Global Control Fuzzy Genetic Algorithms

GUO Hairding, LI Xiangping

(College of Energy and Power Engineering,

Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract:** A global control Fuzzy Genetic Algorithm (FGA) was applied to the structure optimization. The method given here differs from the standard GA methods (SGA) in taking the whole process of Genetic Algorithm selection, crossover, mutation and the change of searching space into global fuzzy control. The quality of FGA method is verified using two typical mathematical multi-peak functions. The results of simulation show that this kind of fuzzy genetic algorithms can efficiently avoid being entrapped into local optimum and obtain a higher efficiency of optimizing process. A rivet structure, which contains integer and discrete variables, was optimized with FGA method. It shows that a better connecting efficiency with less converging steps can be obtained.

**Key words:** aerospace propulsion system; Fuzzy Genetic Algorithms (FGA); structural optimization; global control

随着科技的发展, 现代航空发动机的结构日益复杂。结构优化问题往往是混合连续变量和离散变量、多目标、多约束的情况, 传统的优化方法

难以有效解决此类问题。遗传算法(Genetic Algorithms, 简称 GA)作为一种新的优化技术, 近年来得到了广泛的研究和关注<sup>[1, 2]</sup>。GA 算法对目标函

收稿日期: 2004-06-13; 修订日期: 2004-10-14

作者简介: 郭海丁(1958-), 男, 山东潍坊人, 南京航空航天大学能源与动力学院副教授, 博士, 主要从事工程结构强度、断裂、疲劳损伤及结构优化设计方法等的研究

数和约束的要求甚少, 适宜并行计算, 对特殊问题可灵活地构造启发式进行求解。由于 GA 算法不硬性规定追求唯一的“精确解”, 而用一系列的满意解取而代之, 这些特点使得 GA 算法在求解优化问题时表现出突出的优势, 极大地拓展了结构优化设计的领域。

但在实际应用中, GA 算法仍存在搜索效率较低和过早收敛等问题。针对此问题, Srinivas 提出了“自适应遗传算法”概念<sup>[3]</sup>; Michalewicz 提出了“遗传算法= 数据结构+ 进化程序”的思想<sup>(1994)</sup>, 用模糊逻辑系统控制 GA 算法运行参数的研究近年来也有所增加<sup>[4]</sup>。

遗传算法运行过程中存在很多模糊性的信息, 如“收敛缓慢”、“适值较高”等, 而模糊逻辑系统建模简单、直观、鲁棒性好。有研究表明, 利用模糊控制的遗传算法性能得到了很大提高<sup>[4]</sup>。但是已有研究成果大多着重于对某单一因素的控制。最常见的就是对交叉率和变异率的模糊控制<sup>[5~7]</sup>, 这一定程度上制约了模糊遗传算法达到更好的性能的可能。实际上, 遗传算法中的诸多因素存在交叉影响, 需要综合考虑。为此本文提出一种改进的全程模糊控制遗传算法。该算法将利用模糊机理综合控制遗传算法的选择、交叉、变异及搜索空间等的变化, 使得遗传算法在寻优时, 在跳出局部最优和搜索效率方面均获得了较大改进, 复杂函数数值模拟考核和结构优化算例均证明其有很好的适用性。

## 1 全程控制的模糊遗传算法 (FGA 法) 原理

基于 Holland (1975) 和 Goldberg (1989) 的标准遗传算法 (Standard Genetic Algorithm, 简称 SGA) 的主要特点是: 根据适应度的大小, 通过对一组随机产生的初始解进行交叉、变异、选择等一系列操作, 从而“进化”到最优解或次优解。交叉率 ( $P_c$ )、变异率 ( $P_m$ ) 的初始值在某一范围内随机选取 (一般情况下  $P_c: 0.5 \sim 1.0$ ,  $P_m: 0.01 \sim 0.1$ ), 确定之后在整个遗传算法寻优过程中将不再改变。选择操作以 Holland 的轮盘赌选择法最为著名, 即后代的生存概率将正比于它的适值。SGA 算法将这一过程高度简化不可避免地导致了早熟和易陷入局部最优解等问题。以下从整体的角度对遗传算法的运行过程进行模糊控制, 给出相应的模糊机理和模糊逻辑。具体包括:

### 1.1 全程控制的模糊选择机制

选择机制包括三方面: 采样空间、采样机理和选择概率。SGA 法的后代产生后直接替换掉双亲 (采样空间是规则的、大小是确定的), 决定哪些染色体被挑选出来进入下一代根据的则是轮盘赌概率。染色体的适应值越高, 被挑选出来的概率越大。这样操作的缺点是显而易见的: 因为后代的适值可能不如双亲 (退化), 直接替换掉可能损失某些好的基因; 在进化的初始或者最后阶段, 某些适应值高的染色体完全占据了优势, 从而使得交叉变得毫无意义, 整个进化变成随机搜索。

为了解决因为生存概率正比于适值所带来的以上问题, 本文提出一种模糊的排序选择法: 让后代和双亲都同样参与竞争 (采样空间是扩大的, 大小是不确定的), 依据适值将扩大的采样空间模糊地划分为“好”、“中”、“差”三类, 再在每一类里面按适值的大小顺序各选择均等个数的染色体来组成下一代 (保持种群规模不变)。这样既保证了种群的多样性, 使得染色体之间的适值始终保持比较稳定的差距, 又避免了“好”基因的遗失, 还能加快收敛。

### 1.2 模糊的交叉和变异的全程调整

很多模糊遗传算法根据种群适值情况来对交叉率和变异率进行整定<sup>[5~7]</sup>: 当种群在局部最优趋于成熟时增加交叉和变异, 在解空间发散时减少交叉和变异; 同时根据个体适值和最好适值的差距来调整交叉率和变异率, 使具有高适值的个体减少交叉和变异的机会, 低适值的个体增加交叉和变异。但此类模糊遗传算法无法判断目前是会陷入局部最优解还是接近全局最优解, 两者从适值的变化趋势来看是相似的, 但本质却完全不同。因此本文对交叉率和变异率的调整不仅考虑当前代种群的适值情况 (横向比较), 而且将参考若干代过程中适值的变化情况 (纵向分析)。

### 1.3 模糊的变搜索空间

在遗传算法研究中, 搜索空间问题过去没有得到足够的重视, 这是因为遗传算法的内在机制会使得解空间逐步缩小从而达到最优或次优解。本文将重点研究如何根据适值的变化情况逐步缩小解空间的范围, 同时又避免陷入局部最优解。

由于每一代的个体数目是一定的, 这种逐步缩小搜索空间 (即在有限的空间内始终产生一定量的个体) 可以看成是模糊地改变了种群规模, 即搜索成为逐渐细化的过程。此时, 相对原来的搜索

空间而言,解的数量是逐渐增长的,从而亦使搜索到最优解的可能性大大增加。

## 2 模糊遗传算法的实现

### 2.1 模糊逻辑系统控制原理

一个典型的模糊逻辑系统应该包括如图 1 的几个部分:模糊化过程、推理过程和反模糊化。

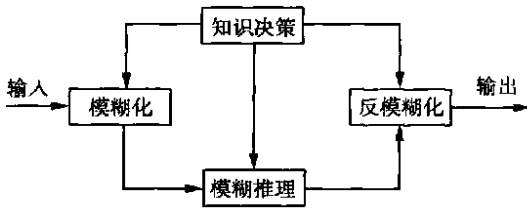


图 1 模糊逻辑系统原理图

Fig 1 A fuzzy logic control system

设计模糊逻辑控制系统第一步是确定输入/输出变量,然后根据变量的范围确定相应的模糊子集及隶属函数,这样输入/输出的语言值就确定好了。图 1 中的知识决策也称为知识库,由领域专家知识或者控制人员实际经验组成的控制规则。模糊推理则利用一定的矩阵运算对模糊规则进行合成,得出的结果经过反模糊化过程最后变成清晰量的输出。本文所使用的模糊逻辑系统均属于该类型。

### 2.2 交叉率的模糊控制

#### 2.2.1 确定输入、输出变量

由 1.2 节的分析可知,对于本文的模糊遗传算法,种群多样性的指标可以简化。设模糊逻辑控制系统的 2 个输入变量分别为  $a$  和  $b$ 。其中:  $a = f_{\max} - \bar{f}$ ,表示当前代的最大适值( $f_{\max}$ )与平均适值( $\bar{f}$ )的差,亦称种群的多样性因子; $b$ 为重复因子。其含义是,将每一代的平均适值与上代相比,若平均适值低于上代,则  $b+1$ ,否则  $b$  的值恒为 1。显然,当  $b$  维持较小的值时,表明种群处于正常的进化过程,对交叉率  $P_c$  的控制适中;若  $b$  逐渐增大,表示种群的进化能力减弱,这时应该避免无谓的交叉, $P_c$  应该较小;反之亦然。 $P_c$  直接作为输出变量(被控制量)。

#### 2.2.2 确定输入、输出变量的隶属度函数

隶属度表示模糊变量隶属于某一模糊集合的“程度”,隶属度函数则代表相应模糊集合的特征。目前基于经验常用的隶属度函数有三角型、高斯型等。本文中,针对  $P_c$  的控制,输入、输出变量的

隶属度函数均采用高斯(Gauss)型。进行如下处理:将适值做归一化处理,则输入变量  $a$  的范围是  $(0, 1)$ ,输出变量  $P_c$  取值也是  $(0, 1)$ 。对于输入变量  $b$ ,由于算法设置为最大代数(200 代)终止,故  $b$  的范围是  $(0, 200)$ 。将  $b$  和  $P_c$  划分为 5 个语言值,即:很小(VS)、小(S)、中(M)、大(B)、很大(VB);对  $a$  使用 3 个语言值,分别为:小(S)、中(M)、大(B);限于篇幅,仅以  $P_c$  的隶属度函数为例,见图 2。

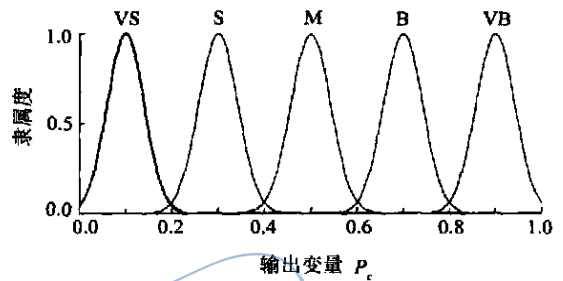


图 2 交叉率的隶属函数

Fig 2 The membership functions of output linguistic variable  $P_c$

### 2.2.3 模糊规则与模糊推理方法

据前分析制定以下 15 条规则控制  $P_c$ ,见表 1。

表 1 交叉率的模糊控制规则表

Table 1 Fuzzy control rules of output linguistic variable  $P_c$

| $P_c$ | VS | S | M | B  | VB |
|-------|----|---|---|----|----|
| S     | B  | M | S | VS | VS |
| M     | B  | B | M | S  | VS |
| B     | VB | B | B | M  | S  |

表中与  $P_c$  同一行的是重复因子  $b$  的语言值,与  $P_c$  同一列的是种群多样性因子  $a$  的语言值。模糊推理方法选择 Mamdani 推理方法。该方法将模糊蕴涵关系用其笛卡尔积表示,即:

$$R = A \times B \times C = (A \times B) \rightarrow C$$

式中, $R$  表示一个模糊关系, $A$  和  $B$  分别是不同论域  $X, Y$  上的模糊集合。 $C$  是对应的模糊结论。

#### 2.2.4 反模糊化方法

反模糊化方法均采用重心法。该方法取模糊隶属度函数曲线与横坐标围成面积的重心作为模糊推理的最终输出值。公式如下<sup>[8]</sup>:

$$v_0 = \frac{\int \mu_v(v) dv}{\int \mu_v(v) dv}$$

其中:  $v_0$  表示模糊系统最终输出值,  $\mu_v(v)$  表示变量  $v$  对应相应模糊集合的隶属度。

与最大隶属度法比较, 重心法结构简单, 推理清晰, 具有更平滑的输出推理控制。

2.3 变异率的模糊控制

变异率的控制类似, 同样是选用一个两输入一输出的模糊控制系统。不同的是输入变量  $b$  记录的是多次未找到更好的最大适值的次数。模糊控制规则见表 2。

表 2 变异率的模糊控制规则表

Table 2 Fuzzy control rules of output linguistic variable  $P_m$

| $P_m$ | VS | S | M | B  | VB |
|-------|----|---|---|----|----|
| S     | M  | B | B | VB | VB |
| M     | S  | M | B | B  | VB |
| B     | VS | S | M | B  | B  |

其中, 与  $P_m$  同行的是重复因子  $b$  的语言值, 与  $P_m$  同一列的是种群多样性因子  $a$  的语言值。输出  $P_m$  的范围是  $(0, 0.4)$ 。

2.4 搜索空间的模糊控制

搜索空间的模糊控制采用单输入单输出模糊控制系统。输入的变量与变异率的重复因子  $b$  相同, 共划分为 5 个语言值。输出变量为比例因子  $S$ , 范围为  $(1, 10)$ , 同样划分为 5 个语言值。模糊规则基于以下经验: 如果种群多次没有找到更好的个体, 则适当扩大搜索空间; 如果种群收敛性能好(重复因子小), 则逐步缩小搜索空间。控制规则如表 3。

表 3 搜索空间的模糊控制规则表

Table 3 Fuzzy control rules of search space

| $b$ | VS | S | M | B  | VB |
|-----|----|---|---|----|----|
| S   | M  | B | B | VB | VB |

2.5 模糊遗传算法流程

所建立的模糊遗传算法的总体结构如图 3。

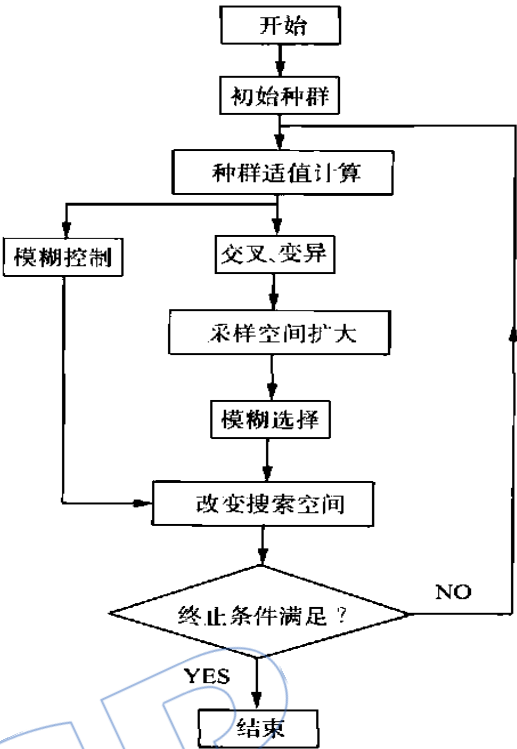


图 3 模糊遗传算法流程图

Fig 3 Flow chart of Fuzzy Genetic Algorithm

3 模糊遗传算法在结构优化设计中的应用

3.1 在数值函数优化中的应用

为考核算法有效性, 选取数学上典型多峰函数 Ackley 函数和 Schaffer's 函数进行数值仿真。这两个函数具有很强的震荡特性, 在全局最优解附近分布着多重局部极值, 一般的优化算法极易陷入局部最优。函数表达式如下:

(1) Ackley 函数

$$f_1(x) = -20 \exp \left[ -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right] - \exp \left[ \frac{1}{n} \sum_{i=1}^n \cos 2\pi x_i \right] + 20 + e$$

$|x_i| \leq 5$

当  $n=2$  时, 该函数有无数个局部极值点, 但是只有一个全局最小点  $(0, 0)$ , 最小值为 0。

(2) Schaffer's 函数

$$f_2(x, y) = 0.5 - \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{[1 + 0.001(x^2 + y^2)]^2}$$

$-100 \leq x, y \leq 100$

表 4 仿真结果数据的统计和比较

Table 4 Statistics and comparison of simulation results

| 函数       | 算法  | 收敛次数 | 平均收敛代数 | 跳出局部极值概率/% | 平均优化值    |
|----------|-----|------|--------|------------|----------|
| Ackley   | SGA | 16   | 120    | 32         | 0.002 30 |
|          | FGA | 32   | 67     | 64         | 0.000 00 |
| Schaffer | SGA | 5    | 101    | 10.7       | 0.990 28 |
|          | FGA | 39   | 82     | 59         | 1.000 00 |

该函数有无数个局部极值点,但只有一个全局最大点(0,0),最大值为1<sub>0</sub>。

以下普通遗传算法用SGA表示,模糊遗传算法用FGA表示。以MATLAB为仿真环境,采用实数编码,初始种群均设为50,每次进化200代,精度设置为1e-5(与目标值差距小于该范围即认为收敛),交叉率 $P_c$ 在(0,1)、变异率 $P_m$ 在(0,0.5)内随机取初始值(SGA的 $P_c$ 和 $P_m$ 分别取0.6,0.1)。 $SGA$ 、 $FGA$ 各自运行50次,统计仿真结果见表4。

从表4中可以看出:在初始条件相同的情况下, $FGA$ 算法的收敛次数要多于 $SGA$ 算法;运行的代数一样的情况下(200代), $FGA$ 算法收敛到的解的质量要优于 $SGA$ 算法; $FGA$ 算法的平均收敛代数少于 $SGA$ 算法;对有的函数 $SGA$ 算法难以保证收敛, $FGA$ 算法却可以比较快地收敛到最优值,并且在陷入局部极值后, $FGA$ 算法跳出局部极值的能力也比 $SGA$ 算法强。

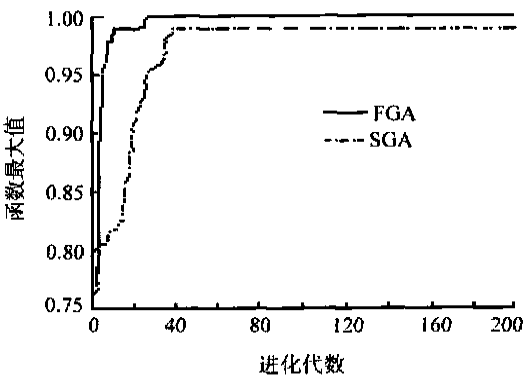


图 4 FGA 算法与 SGA 算法收敛性能比较

Fig 4 Convergence comparison of FGA and SGA

图4是FGA算法与SGA算法对Schaffer's函数的寻优性能比较。其中SGA运行200代的时间在30~60s之间,FGA由于加入了模糊推理计

算,故运行200代的时间稍长,在1~2min左右。由于FGA比SGA普遍收敛快,达到收敛时所用的代数少,从寻找到最优解的时间看,FGA更具优势。从图4看出:SGA算法陷入了局部极值(0.99028)很难跳出,而FGA算法通过若干代的适应和调整能够达到全局最优(1.00000)。

3.2 FGA 算法在结构优化中的应用

航空发动机连接结构设计复杂。合理选择连接件的数量、尺寸以及固定的位置对有效的连接设计十分重要。本文选取一个板连接处结构优化设计作为说明——该算例要求对连接处铆钉的尺寸、数目和布局进行优化设计,是一个混合整数变量和离散变量、多约束的结构优化问题。其中:变量 $x_1$ (铆钉排列行数)要求是1~32之间的整数;变量 $x_2$ (每行铆钉数)取0~128之间的整数;变量 $x_3$ (铆钉直径,mm)则要求是[6,8,10,12,14,16,18,20,22,24,27,30,33,36,40,45]等数值中的一个。优化目标是使连接效率达到最大,其中剪切失效强度( $p_s$ )、拉伸失效强度( $p_t$ )和挤压失效强度( $p_b$ )公式如下:

$$\begin{aligned} p_s &= \frac{\pi x_3^2 x_1 x_2 \tau}{4} && \text{if } x_1 < 3 \\ &= \frac{\pi x_3^2 x_1 x_2 \tau}{4[1.06 + 0.126(x_1 - 3)]} && \text{if } x_1 \geq 3 \\ p_t &= (2\,000 - x_2 x_3) t \sigma_t \\ p_b &= t x_3 \sigma_b x_1 x_2 && \text{if } x_1 < 3 \\ &= \frac{t x_3 \sigma_b x_1 x_2}{1.06 + 0.26(x_1 - 3)} && \text{if } x_1 \geq 3 \end{aligned}$$

上列式中:

$\tau = 80\text{ MPa}$ ,  $\sigma_t = 90\text{ MPa}$ ,  $\sigma_b = 120\text{ MPa}$   
且各变量还需满足下列两个约束条件:

$$3x_3 x_1 + 2x_3 \leq 500$$

$$3x_3x_2 + 2x_3 \leq 2\,000$$

连接效率就是求最小(大)失效强度与边界拉力的比值。连接处几何特性见图 5, 该算例来自文献[8]。

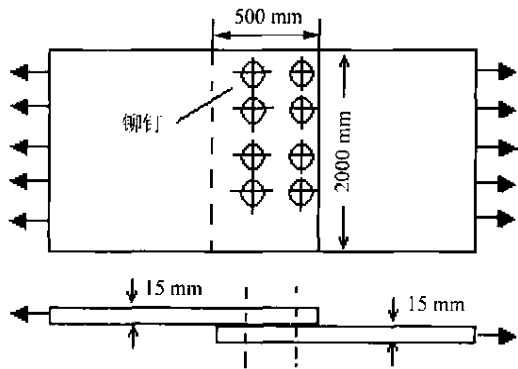


图 5 铆钉连接处结构示意图  
Fig 5 Sketch of riveted lap joint

模糊遗传算法采用与 3.1 节相同的设置, 自然编码方式, 将各约束采用惩罚函数处理(不满足约束条件的解将其搭接效率设为 0), 优化到最好的解是  $x_1=5$ ,  $x_2=15$ ,  $x_3=27$  mm, 最大连接效率达到了 85.44%。两种算法优化结果见表 5。

表 5 板件连接优化设计结果对比

Table 5 Optimum of lap joint efficiency of two methods

| 方法                         | 文献[8]       | 本文          |
|----------------------------|-------------|-------------|
| 编码方式                       | 二进制         | 实数          |
| 收敛代数                       | 58          | 34          |
| 最优解 mm [ $x_1, x_2, x_3$ ] | [5, 13, 27] | [5, 15, 27] |
| 连接效率/%                     | 69.79       | 79.75       |
| 最大连接效率/%                   | 82.45       | 85.44       |

可见与 SGA 算法相比, FGA 算法以较少的收敛代数达到了更高的连接效率。

4 结 论

(1) 通过对遗传算法的选择、交叉、变异以及搜索空间的全程模糊控制, 不仅提高了搜索效率, 而且可以有效地改善未成熟收敛。

(2) 模糊逻辑系统的原理简单、实现方便, 由于充分重视和利用了遗传算法过程中的一些“软”的因素, 达到了改善遗传算法性能的目的。

(3) 本文的方法不论对连续变量或离散变量优化问题均有很好的适应性(数值算例是连续变量, 工程算例是离散变量), 故对于实际工程结构优化问题具有更好的通用性。

参考文献

[1] (日)玄光男, 程润伟. 遗传算法与工程设计[M]. 北京: 科学出版社, 2000.

[2] 崔海涛, 桑韧, 温卫东. 基于遗传算法的连续结构拓扑优化分析[J]. 南京航空航天大学学报, 2004, 36(2): 159- 163.

[3] Srinivas M, Patnaik L M. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms[J]. IEEE Transactions on Systems, Man and Cybernetics, 1994, 24(4): 656- 67.

[4] Herrera F, Lozano M. Fuzzy Adaptive Genetic Algorithms Design, Taxonomy, and Future Directions[J]. Soft Computing, 2003, 7(8): 545- 62.

[5] 王平洋, 王广生, 胡兆光. 加快遗传模糊算法的搜索过程[J]. 电网技术, 1999, 23(5): 3- 6, 10.

[6] 李擎, 郑德玲, 唐勇, 等. 一种新的模糊遗传算法[J]. 北京科技大学学报, 2001, 23(1): 85- 89.

[7] 戚志东, 朱新坚, 朱伟兴. 基于模糊规则优化的改进模糊遗传算法[J]. 计算机工程与应用, 2003, 39(27): 18- 20, 38.

[8] 闻新, 周露, 李东江, 等. MATLAB 模糊逻辑工具箱的分析与应用[M]. 北京: 科学出版社, 2002 年(第二版).

[9] Lin C Y, Hajela P. Genetic Algorithms in Optimization Problems with Discrete and Integer Design Variables[J]. Engineering Optimization, 1992, 19(4): 309- 327.